L2STT monitoring

Sailesh Chopra
SCRI, FSU

The monitoring system intended for the L2STT is supposed to fit seamlessly into the D0 level2.

Various cards in the Level 2 STT crates are:

VBD
- VME Master to read out to L3
- Cannot interrupted during readout
- Must read from same set of VME addresses for every event

Fiber Road Card
- Receive data from the CFT L1 trigger over G-link 16bits at 53 MHz via VTM, one fiber per FRC
- Broadcast the L1CFT data to trigger cards using special purpose bus on the back-plane
- Receive control signals from trigger FW over SCL
- Control buffering and readout to L3
- VME controller

Trigger Cards (9 per crate)- STC
- Receive CFT data over backplane and convert it to roads in SMT.
- Receive SMT data from sequencer over optical fibers and find hit clusters G-link, 16 bits at 53 Mhz via VTM. Four fibers each transmitting data from 2 HDIs per STC.
- Buffer SMT data
- Select *r-phi* hits inside the roads defined by the CFT data
- Transmits hits in roads to TFC.
- Buffer hit clusters for readout to Level 3
- Has 16 event deep buffer. The transfer to TFC starts when there is data in the event buffer.

Track Fit Card (2 per crate)- TFC
- Receives input from FRC over LVDS serial link, 32 bits at 26 MHz.
- Receives input from STC over LVDS serial link, 32 bits at 26 MHz.
- Perform track fit.
- Sends data to L2CTT over cypress hotlink, 16MB/s. One cable per TFC but not more than 2 cables/crate.

- Sends data to L3 over VME bus via VBD.

Catalogue of information to be monitored:

Sources of monitoring information:
- VTM (?)
- Individual cards (FRC/STC/TFC)

Monitoring Options

The control and collection of monitoring information from various pre-processors cards can be done using one of the following options:

1. FRC controls the monitoring i.e. it collects the information from the pre-processors cards and write onto MPM (it is VME controller/bus master) to be read by the TCC.
2. Use dedicated CPU (68k/alpha) to work like Administrator alpha in other L2 crates. This can take care of controlling/processing of the monitoring information and can also be one of the paths for downloading data to the various cards.

The monitoring information will collected after receiving L2collectstatus and transmitted to TCC.

The SVX-VRB style monitoring uses a dedicated CPU card, which at present uses Motorola 64k, but will eventually use either PowerPC or 2400 series CPU. It has following the following characteristics (advantages)

1. One can ask for monitoring information collection and transmission over the VME backplane as frequently as our time constraints and VME bus occupancy allows. But collecting this information every 5 seconds shouldn't be a problem. Since once VBD has the bus control, it will not be possible for any process to use the VME bus, which may give rise to bus contention. This can be avoided by using the P2P link between this CPU card and other cards. VRB group has not yet finalized on the design and the protocol that will be used for P2P connection (firewire or USB) etc. We can as well use PCI interface for communication between the CPU card and other daughter cards…
2. VRB design of the monitoring CPU card has the on board memory of 32 or 64MB to store the data collected for monitoring purposes before it is transmitted to the TCC/host computer. The amount of memory depends on how much monitoring data we have to collect. The onboard CPU is used to compute histogram etc.
3. This card has the Ethernet connection to the host computer, through which it can send and receive data. This card can be programmed using the C language in VxWorks environment.
4. This card can also be used to control the downloading of the information from the host computer directly or through the D0 secondary DAQ path, to the daughter

cards at power up time. This information can be the various LUTs needed by various cards.

5. This card can be used to send directly the warning/error messages to the error logger/alarm system if something weird starts to happen, like broken link or power supply.

For monitoring purposes, what we may have to do is to store the monitoring information in some registers/buffers addressable by this CPU/VME on all daughter cards. When FRC receives the SCL collect_status signal, it lets the CPU card know to collect the information from the daughter cards before transmitting it to the monitoring host/TCC. If we use the VME backplane to for the monitoring information, then additional mechanism has to be made available to let CPU cards known when VME bus is available and when it is done it should let FRC know that it is done. With P2P connection, the information can be collected independent of the status of the VME bus.
The disadvantage of this card is the addition of one more card to the STT. But since this card is already (being) designed for the VRB monitoring most of the work is already done. All we have to is to integrate it in the STT system and re-program it, for which the expertise is available with in D0.

The FRC card can also be used to collect and transmit the monitoring information from various daughter cards. Since it is the bus master for the STT cards, it always knows the state of VME bus and controls the flow the events through various daughter cards. The drawback of using the FRC card for monitoring also is that one loses flexibility for information manipulation/ downloading will be reduced. And it will add another layer of complexity to the FRC design. Also without the CPU card the direct connectivity of the STT crate to the rest of the world via Ethernet will not be available.

Menu of Information to be Monitored for Individual Cards.

For collecting the monitoring information, the FRC card is used as an example.
The contents of the monitoring register will be updated on each event, and will be reset after each *collect_status* request by writing to the last word in the memory block.. The three broad classes of monitoring data made available to monitoring registers are:

- *Event:* a snapshot of the current value of the data for the particular event
- *Accum:* a running sum of the data for each event since the last reset of the monitoring registers
- *Histo:* a histogram of the data filled for each event since the last reset of the monitoring registers.

FRC Monitoring:

All of the following information is taken from the FRC document...

The monitoring registers in the TFC will be implemented as a contiguous memory space of 32-bit words, it is expected that 256 bytes will be sufficient.
The following tables indicate various monitoring registers and the brief description for the various sections of the FRC card. For complete description of these words/registers refer to the TFC document.

SCLF (SCL formatting & Control) Monitoring Register			
Offset(hex)	*Name*	*Class*	*Description*
00-1c	**L1_QUAL**	Histo	Histo of L1 qualifier bits
20	**L1_PERIOD**	Accum	Number of L1 periods
24	**L2_PERIOD**	Accum	Number of L2 periods
28	**L2_ACCEPT**	Accum	Number of L2 accepts
2c	**L2_ACCEPT**	Accum	Number of L2 accepts
fc	**RESET**	Write	Reset register Contents

Road Receiver (RR) Monitoring Register			
Offset	*Name*	*Class*	*Description*
00	RR_OPTPOW	Event	VTM Finisar receiver optical power
04	RR_CABLE	Event	G-Link cable connected
08	RR_RESET	Accum	Number of L2 resets to RR
FC	RESET	Write	Reset register contents

Buffer Manager Monitoring Register			
Offset	*Name*	*Class*	*Description*
00	BM_NFREE	event	Number of free buffers
04	BM_NUSED	event	Number of used buffers
08	BM_WRITE	event	Current buffer # for write
08	BM_READ	event	Current buffer # for read
08	BM_INIT	event	N SCL init's broadcast
08	BM_VBD	event	N(cycles) w/ VBD hogging VME
FC	RESET	write	Reset register contents

STC Monitoring:

For STC two type of monitoring information is anticipated... one of these is an SMT error count indicating the number of times the SMT data was received with an incorrect header, thereby causing the strip reader to resume searching for a new header.
The other monitoring data is an 8-bit histogram of channel hits (occupancy) which is a count of number of times the data is received for the channel. Since there can be maximum of 9-chips per input with 128 channels/chip, it will require a total of 1152x8bits of storage for each input. Along with these there can be other items which may have to be monitored like:

- Average buffer/FIFO occupancy (or histogram of fraction of time with 0...16 events)
- Time spent in each processing state, which will need a slower clock (.1 to 1KHz), it will have to be designed into the cards
- Size of data transferred…
  - Average size of clusters
  - Average # of clusters per CTT road
  - CTT tracks per sector/event

In addition to these provision also have to be made for sending unbiased data sample from input of STC to level3 along with processed data (clusters and clusters associated with roads). The STC can transmit following information to L3.

- All the inputs from FRC
- All the inputs from SMT and
- All the axial and stereo clusters

TFC Monitoring:(From TFC document)

Monitoring of the input and output buffers will be provided on the receiver cards. Six CPU states will be defined: idle, input bus request, input data transfer, processing, output bus request and output data transfer. These will be monitored for all CPU's by the I/O controllers. The states will be latched at fixed time intervals into a single register and read out when requested. This will allow a stochastic sampling of the CPU states. In addition, monitoring internal to the DSP's will include

- Hits overflow count
- Layer overflow count
- Empty CTT track count
- Intermediate calculation results for diagnostic readout

These will be written to the output DPM buffer upon receipt of appropriate SCL qualifiers (from the FRC). All monitoring information will be accessible from VME.

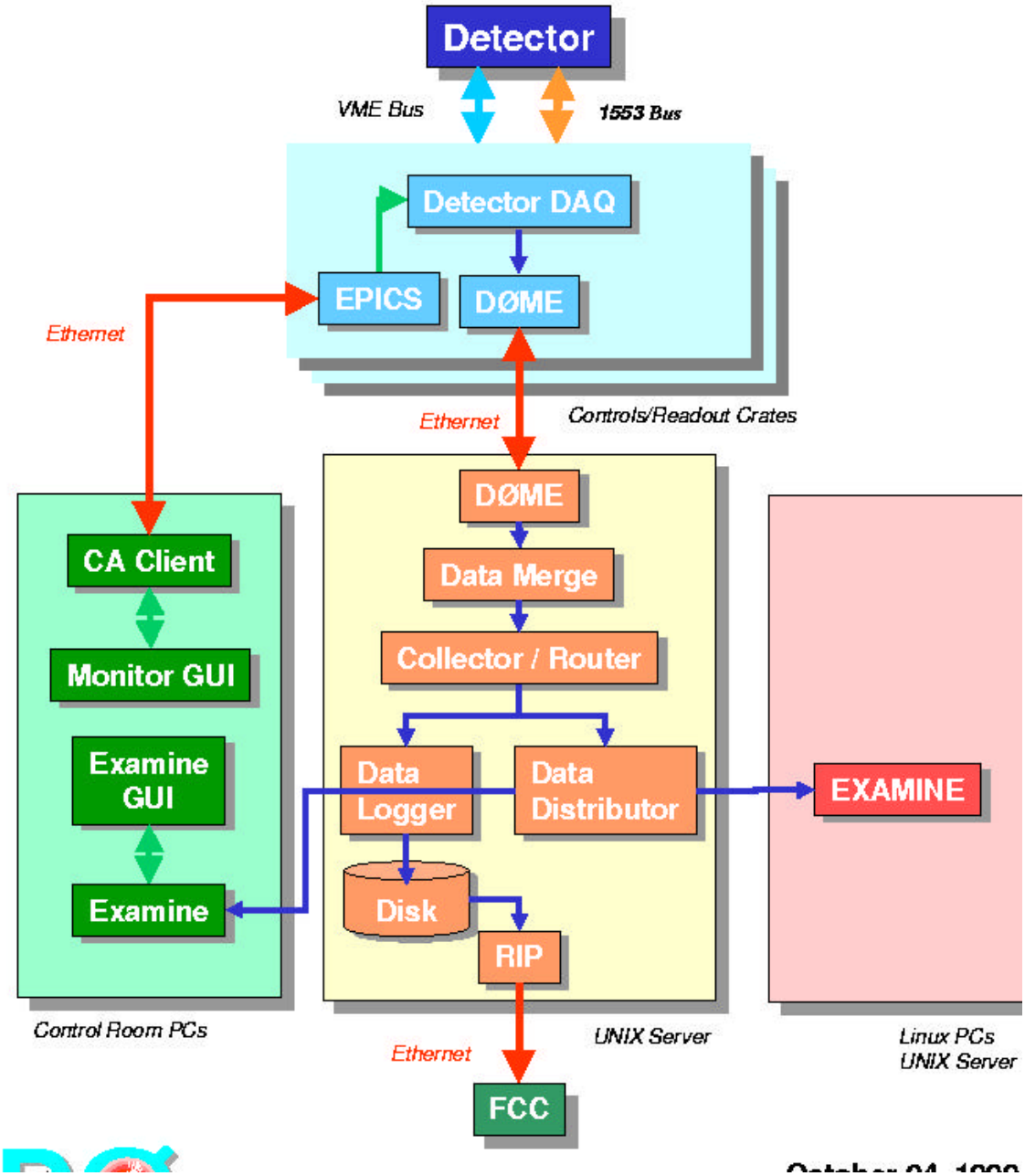
Along with these, for unbiased events TFC will send all inputs (from L1CTT and STC) and all created data (TFC, L2STT) to level 3.

Downloading:

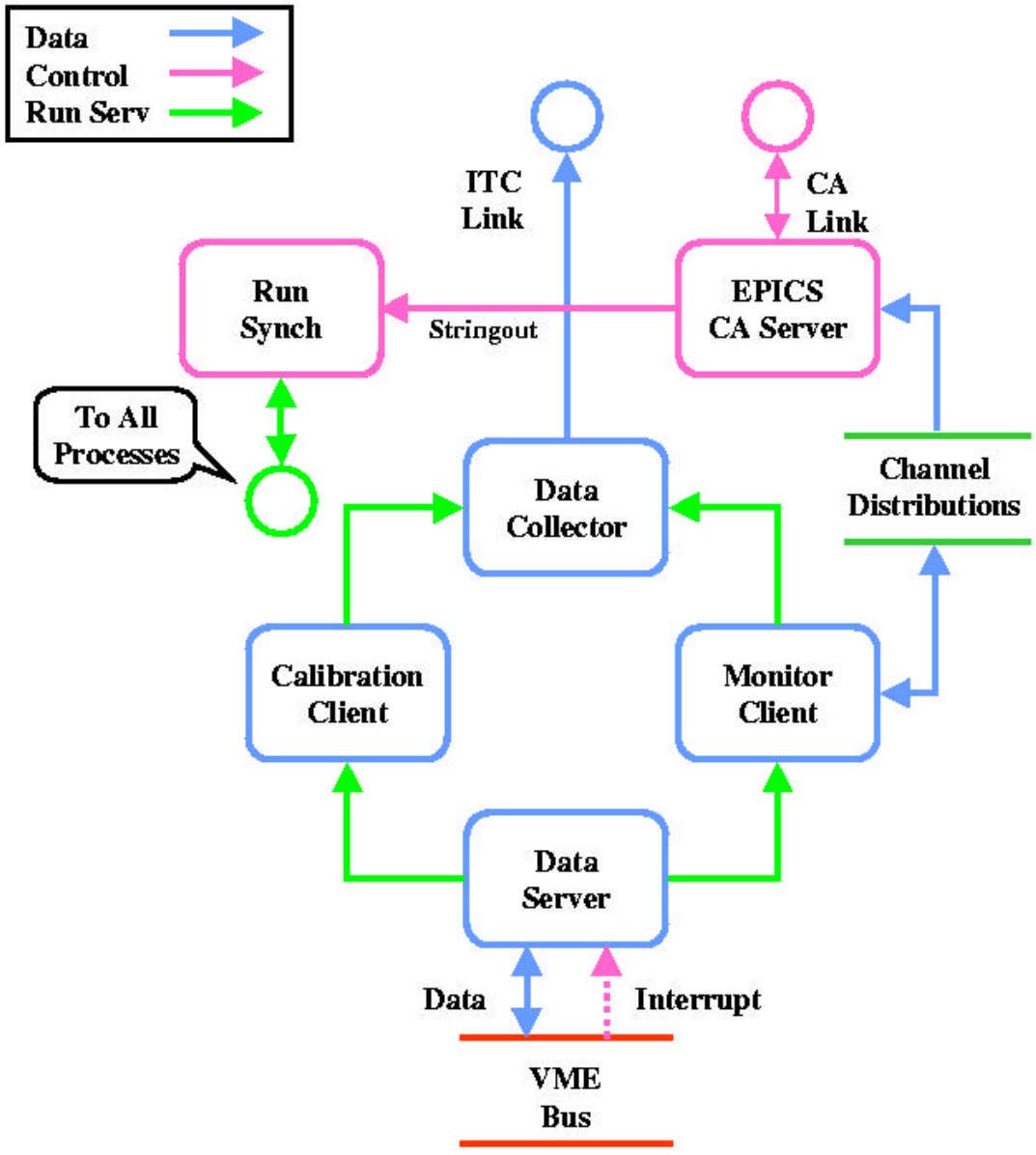
The CPU card described above along with acting as monitoring information server for STT can also be used for downloading the information to various daughter cards. This information includes downloading of pedestals for SMT, Look up tables, run time constants (beam and vertex positions etc). For the purpose of downloading the pedestal, lookup tables one can use the D0 secondary DAQ path to simplify the book keeping, as it takes care of version control etc and the Hardware database. For run time constants, TCC framework can be used...(I am not too sure about these options but writing these anyway to have some starting point for discussions). The D0 secondary DAQ schematics for data flow and Control and monitoring are appended here.

I will try to get the technical details/requirements of this VRB monitoring card for more detailed understanding of the card and issues involved.

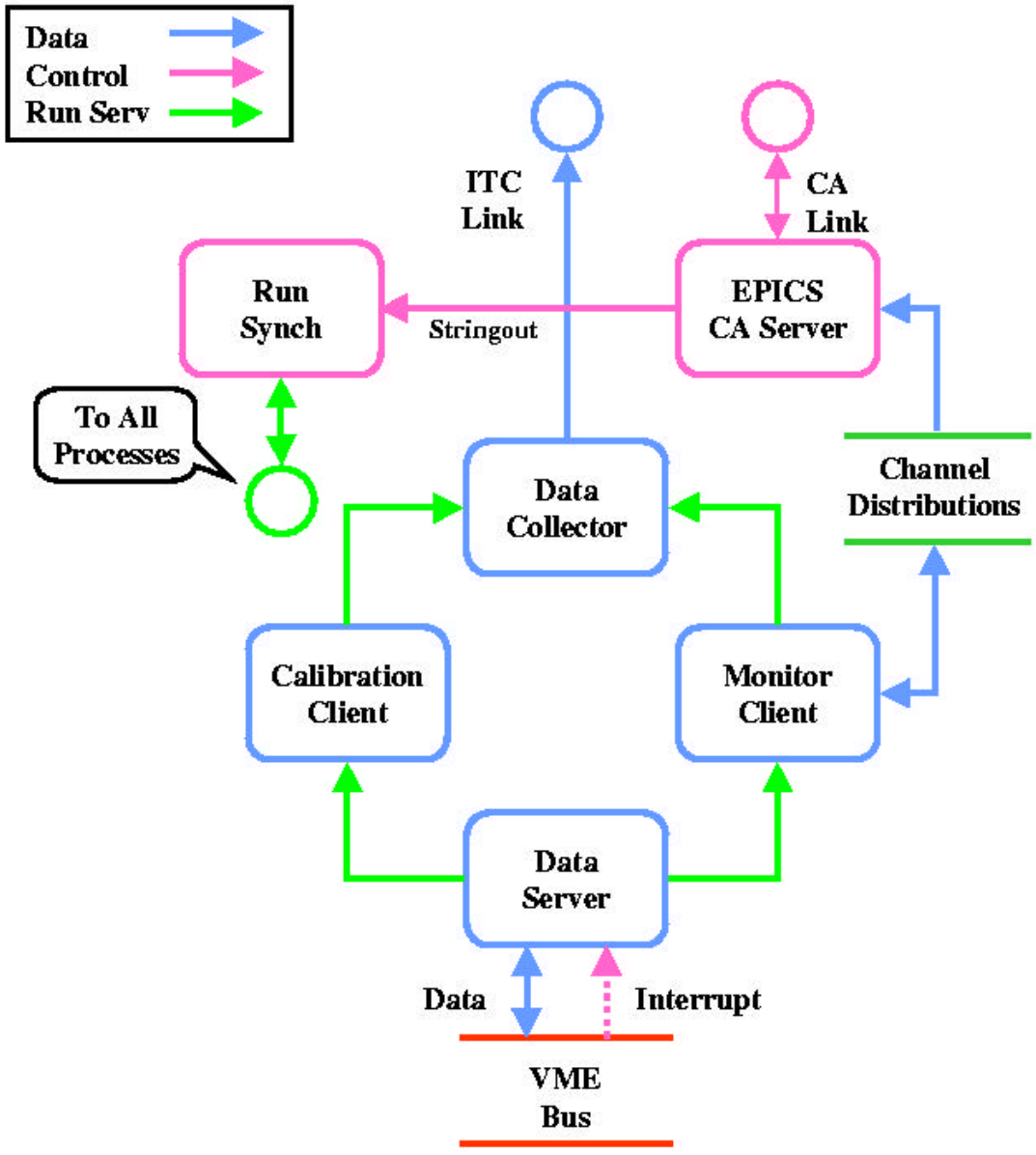
New Secondary Data Flow



New IOC SDAQ


Data
Control
Run Serv
ITC
Link
CA
Link
Run
Synch
Stringout
EPICS
CA Server
To All
Processes
Data
Collector
Channel
Distributions
Calibration
Client
Monitor
Client
Data
Server
Data
Interrupt
VME
Bus

New IOC SDAQ


Data
Control
Run Serv
ITC
Link
CA
Link
Run
Synch
Stringout
EPICS
CA Server
To All
Processes
Data
Collector
Channel
Distributions
Calibration
Client
Monitor
Client
Data
Server
Data
Interrupt
VME
Bus